We Claim:

1. A method for processing a Java Archive (JAR) file to provide an interpretable application file adapted for a target environment, comprising:

removing from said JAR file at least a portion of information not necessary for executing said application;

mapping at least one of application defined interface, class, field and method names to shorter names; and

mapping at least one of target environment defined interface, class, field and method names to corresponding target device names.

2. The method of claim 1, wherein said step of removing comprises:

removing unnecessary byte codes from said JAR file.

3. The method of claim 1, wherein said step of removing comprises:

removing at least one of private unreferenced methods and fields from said JAR file.

4. The method of claim 1, further comprising:

identifying within said JAR file instances of duplicate strings; and

remapping each duplicate string to a corresponding initial string.

5. The method of claim 1, further comprising:

identifying within said JAR file instances of strings;

providing a table to hold one instance of each identified string; and

remapping each identified string to a corresponding string table entry.

6. The method of claim 1, further comprising at least one of the following steps:

(a) removing unreferenced constant pool entries for at least one class;

(b) mapping constant pool entry names to fixed length names; and

(c) sorting constant pool entries by type.

7. The method of claim 1, further comprising:

preferentially remapping application references to at least one of target environment defined interface, class, field and method names.

8. The method of claim 1, wherein:

a target environment obfuscation is provided in which symbols used in the target environment are replaced with shorter names.

9. The method of claim 1, wherein:

an application obfuscation is provided in which symbols used in an application are replaced with shorter names that do not overlap the names used for target environment obfuscation.

10. The method of claim 1, further comprising:

mapping constant pool entry names to names having a fixed length.

11. The method of claim 10, further comprising:

moving strings from the constant pool to a common string pool.

12. The method of claim 1, further comprising:

assigning a global name to at least one of application and target environment methods of each interface class.

13. The method of claim 1, wherein:

said mapping steps are only used for mapping private symbols.

14. A method, comprising:

removing at least a portion of at least one of non-critical archive information, class information and unreferenced member information from a Jar file including an application;

replacing at least one of interface, class, field and method names with corresponding shorter interface, class, field and method names;

replacing at least one of target environment defined interface, class, field and method names with corresponding target device interface, class, field and method names.

15. A method, comprising:

iteratively resolving application defined and target environment defined class, field and method names to interpret application byte codes presented within a ground Jar file.

16. A signal bearing medium including a representation of software instructions which, when executed by a processor,

perform a method for processing a Java Archive (JAR) file to provide an executable application file adapted for a target environment, comprising:

removing from said JAR file at least a portion of information not necessary for executing said application;

mapping at least one of application defined interface, class, field and method names to shorter names; and

mapping at least one of target environment defined interface, class, field and method names to corresponding target device names.

17. A computer program product, comprising a computer data signal embodied in a carrier wave having computer readable code embodied there in for causing a computer to process a Java Archive (JAR) file to provide an executable application file adapted for a target environment, said computer process comprising:

removing from said JAR file at least a portion of information not necessary for executing said application;

mapping at least one of application defined interface, class, field and method names to shorter names; and

mapping at least one of target environment defined interface, class, field and method names to corresponding target device names.